

Модель памяти Java

1. Особенности оригинальной модели памяти Java
2. Синхронизация потоков и видимость переменных
3. Порядок операций в многопоточном приложении
4. Отношение happens-before

- Atomicity (атомарность)
- Visibility (видимость)
- Ordering (порядок)

Начальные значения: $A = B = 0$

Thread 1	Thread 2	Thread 3
$A = 1;$	$\text{while}(A \neq 1);$ $B = 1;$	$\text{while}(B \neq 1);$ $r1 = A;$

Какие значения могут быть для $r1$?

Thread 1	Thread 2	Thread 3
A = 1;	while(A != 1) ; B = 1;	while(B != 1) ; r1 = A;

Какие значения могут быть для r1?

- 1) r1 = 0
- 2) r1 = 1

Thread 1	Thread 2	Thread 3
A = 1;	while(A != 1) ; B = 1;	while(B != 1) ; r1 = A;

- 1) r1 = 0 - Верно
- 2) r1 = 1 – Верно

Возможно, что:

- Thread 2 видит write A первым потоком
- Thread 3 видит write B вторым потоком ПЕРЕД тем как увидит write A

Начальные значения: $A = B = 0$

Thread 1	Thread 2
$r1 = B;$ $A = 1;$	$r2 = A;$ $B = 1;$

Какие значения могут быть для $r1$ и $r2$?

Thread 1	Thread 2
<code>r1 = B;</code> <code>A = 1;</code>	<code>r2 = A;</code> <code>B = 1;</code>

Какие значения могут быть для r1 и r2?

- 1) `< 0, 0 >`
- 2) `< 0, 1 >`
- 3) `< 1, 0 >`
- 4) `< 1, 1 >`

Thread 1	Thread 2
r1 = B; A = 1;	r2 = A; B = 1;



Reordering

Какие значения могут быть для r1 и r2?

- 1) < 0, 0 > - Верно
- 2) < 0, 1 > - Верно
- 3) < 1, 0 > - Верно
- 4) < 1, 1 > - Верно

1. В чем назначение модели памяти в Java?
2. Какие особенности у отношения happens-before?
3. Назовите основные свойства модели памяти и в чем их смысл.